MASTER OF SCIENCE
IN ENGINEERING

Module Description

Embedded Real-time Software

General Information

Number of ECTS Credits

3

Abbreviation

TSM_EmbReal

Version

31.3.2017

Responsible of module

Hans Buchmann, FHNW

Language

	Lausanne	Bern	Zürich
Instruction	☐ E ☑ F	☐ D ☐ E ☐ F	☐ D ☑ E
Documentation	☑ E ☐ F	☐ D ☐ E ☐ F	☐ D ☑ E
Examination	☑ E ☑ F	☐ D ☐ E ☐ F	☐ D ☑ E

Module category

☐ Fundamental theoretical principles

☑ Technical/scientific specialization module

☐ Context module

Lessons

☑ 2 lecture periods and 1 tutorial period per week

Brief course description of module objectives and content

Embedded Systems, although they are not visible, they have become integral parts of this world. Embedded Systems essentially consist of two components, hardware and software. In contrast to information systems in the banking world, hardware is more application specific. Due to this fact, the software that interacts directly with the hardware is more specific as well.

Real-time and parallelism are important issues in Embedded System development, which come on top of the generally valid requirements for correctness and reliability.

The module teaches methods to develop Embedded System Software and deals with the following two complementary aspects:

- Embedded Programming (Programming close to hardware)

- Abstract Modeling Concepts. Both parts are based on Object-Oriented Concepts.

Aims, content, methods

Learning objectives and acquired competencies

Based on requirements, the students will be able to apply the optimal method to develop and verify an Embedded System, covering software on the boundary between hard- and software using C++, as well as the application layer using modeling methods.

Contents of module with emphasis on teaching content

In the first part, the focus is on Near-Hardware-Programming, we use a typical (small) system on chip. The programming language is C++, the programming environment is Linux.

- Hardware-Access
- Interrupts
- Parallelism

In the second part, the focus is on Modeling, a model driven approach: from requirements, over modeling to the running system. □

- Software-Architecture
- Modeling extended Unified Modeling Language (UML)
- Testing of executable Models
- Real-Time Scheduling

Teaching and learning methods

- Ex-cathedra teaching
- Exercises
- Self-study (study of papers, case studies)

Prerequisites, previous knowledge, entrance competencies

- Programming language C/C++
- Computer architectures
- Fundamentals of Operating Systems

Literature

Assessment

Certification requirements for final examinations (conditions for attestation)

None

Written module examination

Duration of exam :	120 minutes
Permissible aids:	Personal summary